



Unstructured source artifacts ontology for requirement validation in GDSD

Sohaib Altaf ¹, Asadullah Shah*², Najma Imtiaz ², Abdul salaam Shah ³, Syed Faiz Ahmed ³

¹Foundation University Rawalpindi Campus, Pakistan

²International Islamic University Malaysia

³Universiti Kuala Lumpur

*Corresponding author E-mail: asadullah@iium.edu.my

Abstract

Analyst while transformation, may not capture true representation of user opinion thus creating deviation from actual user needs. Role of stakeholder is primary in both requirement elicitation and validation process. In elicitation stage they are initiator whereas in validation stage they are authenticator of their needs. Invalid requirements if verified may not be a deviation from specified behavior but deviation from user needs. Requirement validation process ensure that requirements should be representing all primary and intermediary artifacts as well a shared opinions of their contributors. This highlights the importance of source artifacts without which requirement validation cannot assured. Researchers didn't propose any framework for requirement validation in context of GDSD in which source artifacts grows exponentially, and stakeholders are located in diversified locations. Due to unstructured nature of source artifacts, they are not electronically indexed and versioned controlled and this makes difficulty in preserving them. Requirement validation is not possible when source artifacts are not available along with their contributors. The major contribution of this paper to address problem of requirement validation in GDSD in absence of stakeholder to make it collocating environment. This paper introduced ontological solution of requirement validation in which target artifacts are validated against source artifact using their semantic relationship. This paper proposed an unstructured source artifact based ontological model for requirement validation in GDSD environment.

Keywords: Pre-Requirement Specification Traceability Pre-RST, Globally Distributed Software Development GDSD, Requirement Validation, Requirement Artefacts.

1. Introduction

Requirement problems contributes a larger part in the software economy [1, 2]. Maintaining Pre-RST is gaining Requirements engineering elaborate customer's needs through knowledge discovery, communication, negotiation, validation and documentation. It is divided in requirements development and requirements management in which requirements development is process of eliciting, analyzing, specifying, and validating the requirements whereas requirement management comprise of change management and traceability [7, 26]. Requirement elicitation is first step whereas requirement validation in each iteration, is the last step to ensure if written requirements are valid and will satisfy customer needs [26]. Quality requirement should be concise, unambiguous, understandable, identifiable, readable, traceable, prioritized, organized, modifiable, verifiable, useable and ultimately feasible and useable [28]. Valid requirements should be readable, verifiable, traceable, unambiguous, consistent, correct, and complete [8, 27].

Project goals are desired outcome of any software system whereas Objectives are precise targets to achieve goals [17]. Invalid requirements if verified may not be a deviation from specified behavior thus deviation from customer needs and project goals. However any deviation from customer needs create defective software project that cannot be validated in it operational environment [18, 29]. This propagation of defects into later phase of software development results in major cost impact as statistics shows that requirements contribution due to validations defect are

proportionally high; however improving these defects play a significant role in success of software projects [9].

"Requirements are not butterfly that we can pin up in a cabinet" a famous quote describing phenomena of requirement dynamics. Requirement elicitation is a difficult process because it created something from nothing. It deals with many source artifacts and creates various intermediate artifacts. Source artifacts can be any unstructured text such as memos, notes, demonstration feedback, audio & video demonstration, prototypes, blogs, product evaluation notes, project comments, suggestions, various other types of documents etc., and belongs to past projects archives or located in www. Intermediate artifacts can be both unstructured and semi-structured. Example of unstructured contents are i.e. new questions, interview results, observations, feedback collected in different meetings like brainstorming or workshop sessions, change notes etc. These information are collected in textual, audio or video format. Whereas example of semi-structured contents are goals, scope, stakeholder needs, draft statement of requirements, various solutions, etc. [11, 20, 22 and 26].

Role of users stakeholder is primary in both requirement elicitation and validation process when they collaborate and work as contributor of various unstructured contents in initial phases as well as they participate in validation of structured contents in later phases. In elicitation stage they are initiator of their needs and in validation stage they are authenticator of their needs. Different elicitation techniques are designed and applicable in different situation and for various type of stakeholders [27]. Output of requirement elicitation process is an initial draft of requirements that need further refinement and filtration in order to be consistent and

complete as well as aligned with goal and scope [12, 20, 22 and 26]. Analyst in a process of elicitation at this stage, generates various questions for stakeholders in alignment to goals. In an ideal situation much refined questions help in aligned investigation of unstructured contents and therefore result in valid requirements.

1.1. Problem of requirement validation

In practice many requirement problems are due to wrong translation and transformation of requirements from one state to another [20]. Analyst while transformation, doesn't not capture true representation of user opinion thus creating deviation from actual user needs. Any deviation throughout this process results in deviated and invalid specifications that would lead to invalid software solution. In different requirement models validation is last step to filter such problems and to ensure that valid specification is not deviated from, and should be representing, all primary and intermediate artifacts as well as opinions of their contributors [7, 8, 11, 26, 27, 42]. Requirement validation also ensures shared understanding of various users groups. This highlights the importance of source artifacts and their contributors without which requirement validation cannot assured. See figure 1.1 and 1.2 for process of validation.

1.2. Source and target artefacts

This initial draft further examines during requirements analysis process, and evolves and refines through negotiation and validation [12, 26], and thus represents as finished artifacts i.e. requirement specification, business requirements, market requirements, project vision, software scope, use cases or user requirements specification, external interface requirements, concept of operations document etc. [7]. These finished artifacts are version controlled documents and hence represents as structured knowledge. Finished artifacts are developed from unstructured contents after series of transformation from source artifacts and intermediate artifacts [20]. In geographically distributed software development GDSD, requirement engineering is key challenge due to discovery and intensive communication of knowledge from diversified sources, and across many stakeholder groups representing different geographic location and are having varied background. [23]. During elicitation in GDSD, unstructured & semi-structured contents representing both primary and intermediate artifacts, grows exponentially [11, 20 and 29] and further participates into development of finished artifacts. Thus becomes source contents for finished artifacts [20]. With high frequency of source artifacts, it is difficult to manually preserve the sources, and another challenge is to maintain their association with finished artifacts because of large repository of knowledge in remote project sites. Therefore both the primary and intermediate artifacts representing unstructured or semi-structured knowledge are not versioned controlled, and therefore not indexed electronically in automatic manner [11, 22 and 25].

1.3. Problem setting

The requirement development cycle starts from unstructured contents and follow through development of goals, objectives, needs, scope, and other primary & intermediate artifacts, and in series of transformation finally creates the finished artifacts. In order to validate the specified requirements and ensure they are not deviated, all source and intermediate artifacts should be accessible along with their contributors. As discussed in previous section that in GDSD, source contents grows with high frequency and diversity, and even not indexed and versioned controlled [10, 11, 22 and 25]. Without it validating SRS with its source artifacts is not possible when sources are either not preserved or cannot associated and its contributors are either not available or are not associated with.

1.4. Research goal

The goal of this research is to automatically validate the requirements in GDSD. Various efforts are reported in literature in requirement validation in decade of 90's [24, 25, 26, 27]. GDSD evolves in decade of 2000's however no work is addressing issues of automatic validation of requirement in GDSD [22, 23 and 29]. Many studies shows that ontologies are promising and appropriate to support several aspects of requirements engineering. A few discusses the ontological solution of requirement validation [13, 14, 15 and 16] yet fails to address the above goal. Next section describes the main research question. In order to address it, three sub questions are further raised answering which will address the main research question.

2. Research question and methodology

In order to achieve research goal, main question is formulated in this section. In order to narrow down the main research question, further two sub questions are raised.

2.1. Main question: How to validate requirements automatically in absence of distributed stakeholders to make a collocating environment?

Sub Question No. 1: What are existing solutions already exist and what are their limitation?

Sub Question No. 2: How use of ontology can minimize the limitations of existing solution for GDSD?

3. Literature review

3.1. Sub question no. 1: What are existing solutions already exist and what are their limitations?

Verification and validation are independent process to ensure that requirement artifacts should meets the user's perceptions and fulfills the intended purpose of software. Requirement verification is more process-oriented in which structured contents are checked against some formal or structured criteria whereas requirement validation is more user-oriented in which structured contents or SRS is evaluated from unstructured contents or raw requirement artifacts in presence of user, see figure 1 [29]. Figure 2 shows that structured and unstructured contents and their relationship. It highlights the role of analyst in which he is involved with distributed stakeholders in requirement validation from source artifacts. Without maintaining association between both of these artifacts, requirements cannot be validated. Valid requirements may lead to valid software that will satisfy user purpose however verified requirements may or may not be valid requirements [18, 29]. Various opinion & feedbacks are contributed and shared by users in form of unstructured contents during the elicitation phase and analyst have to further transform it in requirements that are technical and more structured [11, 24]. User have to validate these technical requirements however he cannot validate the structured contents and formal models as it is not well understood by him. Analyst cannot validate the technical requirement without unstructured contents shared by user or in absence of user. In order to avoid any deviation from actual user needs, analyst have to validate technical requirement from the user as well as it should be conforming to unstructured contents [24].

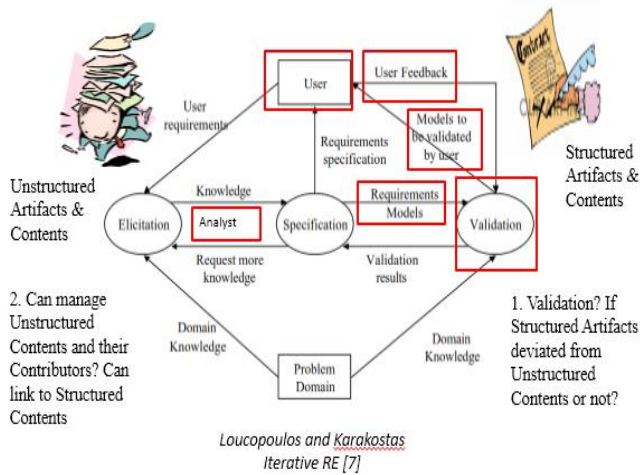


Fig. 1

Traditionally requirement elicitation, analysis, negotiation and specification are human job, however many automated tools are supporting it in decade of 90's, whereas scope of these technological advancement was limited to informal retrieval and data mining [24, 25, 27, 33]. In decade of 2000's new areas are explored for their application in requirement engineering; these areas are artificial intelligence, library science, information science, cognitive psychology, linguistics, statistics, and mathematics. Many tools that are based on natural language process and linguistics techniques, are developed for requirement validation [11, 22 and 29].

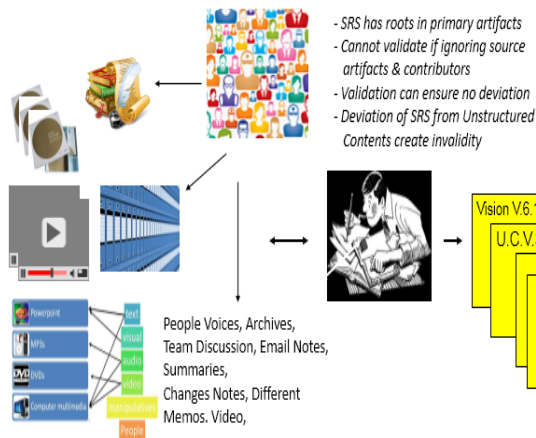


Fig. 2

Many tools are specifically designed for automating requirement elicitation process. In order to understand the automation of this process, it requires to understand its inherent nature. Output of requirement elicitation process is initial gathered information that is expressed in natural language artefacts however this is not well appropriate for system definition because of its inherited ambiguity and incompleteness i.e. repetition of inputs, different jargons, social & linguistics problems. These informal contents are represented in different artefacts of textual, graphical, audio or video format and collectively these represent the valid voice of user and customer, although any single or few artefacts are not complete and similar information is repeated in source artefacts. Many approaches are developed to represent the natural language expressions in structured and formal way in order to reduce ambiguity but user cannot decode and validate these structured contents. Thus process of validation is yet depended on analyst knowledge. Analyst with the help of user facilitate the process of transformation of natural language artefacts in formal artefacts. However it is difficult to validate the equivalency of developed requirements with unstructured contents in absence of user because finished requirements are represented by and associated with repeti-

tive and ambiguous statements in source artefacts. As it is reported in literature that requirement validation is user-centred tasks. Thus source artefacts are used as a reference for later validation with users [24].

Ambriola and Vincenzo [24] proposed a web-based tool to support elicitation, analysis and validation tasks. This tool converts that unstructured contents in various models described by requirements that are semi-structured contents and even understood by user. This tool use various information extraction and mining techniques to extract domain information and abstract of what is shared by user and then convert it in semi structured format that is understandable by user. Assumption of this tool is that user can validate the semi-structured contents easily as compare to structured and technical artefacts. Apparently this looks good in case of user and analyst are at same place however requirements evolves with passage of time and artefacts grows exponentially in GSD and if it is not managed, it will be difficult to trace the developed requirements with its source artefacts as well as its source user who shared it [24]. Many other researchers discussed the problem of validation however few proposed some solution also, but yet fail to address this issues for the GSD [24, 25, 27, 33, 34, 35, and 36].

Research is focusing on improving validation process by providing user semi-structured contents in form of textual or semi-formal models, animation, simulation or any other format that is understandable by user [29]. Concept maps are used for different purposes however in another paper [30] concept maps are proposed for requirement validation also in which requirements are linked with source contents and rationales and both are represented by concept map. The assumption taken here is that analyst may find traces of requirements with its rationale and then validate the requirements. Many tools are proposed for creating such visualized models [31]. However creating such associations and traces is problem of pre-requirement specification traceability pre-RST which is problematic area yet [10, 30]. This paper only guides about usage of visualization model for, and emphasized its importance for validation purpose. This paper does not address the problem of pre-RST without which is difficult to manually create such links. Even it does not discuss about semantic of association without it is difficult to validate requirements in absence of user. Creating manually links between requirements and its sources may create many problems such as it is much time consuming job, finished requirements may be linked with many sources and it is difficult to manually probe those.

Many researchers emphasize the importance of maintaining pre-RST for requirement validation [10, 30, and 37]. Gall & Berenbach [32] emphasize on maintaining rationale of requirements and maintain traces between finished requirements i.e. use case and its source i.e. unstructured contents. Moreover focus of this paper is to maintain rationale and inputs in video format instead in text format. This paper proposed new framework that maintain the meeting record in multimedia formats and link it with finished artefacts. Moreover this paper discussed different such type of tools in their related work. Further this paper pointed out the pre-RST of these multimedia inputs with finished artefacts for requirements validation purpose. But this paper not discussed the problem of pre-RST as well as not addressed the requirement validation process in GSD environment.

Botzenhardt, Witt, & Maedche [11] proposed a text mining tool that mine and extract the voice of customer from unstructured contents and other input documents, for purpose of developing requirements and finally create some semi-structured contents that represents the voice of customer. Authors pointed out that input documents represent true customer inputs and this tool processes the large repository of input documents automatically thus replacing human job of developing requirements. In 2000's many useful contents are available at www. Social media is also useful source for blogs and feedbacks. Text mining tool is powerful to mine many useful pattern from repository of inputs. Focus of author is not to replace analyst, instead process the input document and create some structured contents. This paper doesn't focused on

requirement validation, but this is more useful effort as compare to earlier models and can be used for validation also. Again it is not replacing users and not addressing GSD. In order to precisely validate requirement.

Meth, H., Brhel, M., & Maedche [22] emphasized on automated requirement elicitation process. According to author dealing with manual requirement elicitation is time consuming process because of huge number of inputs and unstructured documents from customers. Author evaluated the effort of automation in area of requirement elicitation, according to which some tools are automated and some are semi-automated. Many automated tools for requirement elicitation is also useful for requirement validation. However much work is required in area of requirement validation in case of GSD.

In case of GSD requirement validation is important process for achieving shared understanding of stakeholders however effective validation is possible in case of communicating with appropriate stakeholder as well useful to address problem of tacit [23, 29]. Communication and collaboration with cross-functional stakeholders groups is a key challenge in GSD due to their diversification and geographically distribution. It is difficult to deal with various artefacts and high frequency of requirements with diversified stakeholders representing various groups, different culture and inter organizational structures [23, 29]. Many papers address the requirement validation process as discussed in previous paragraphs, however none of these works in GSD where maintaining large repository of source artefacts and associate it with finished requirements is problem of pre-RST. Moreover none of these are maintaining relationship of contributors with their source artefacts. Without maintaining these traces and links semantically, it is not possible to automatically validate requirements in absence of distributed stakeholders. In next section we will discuss the usefulness of semantics for purpose of requirement validation.

3.2. Sub question no. 2: How use of ontology can minimize the limitations of existing solution for GSD?

Many studies highlights the use of ontologies for managing knowledge in requirement engineering phase. The review results show that ontologies are promising and appropriate to support several aspects of requirements engineering. Various studies [1, 2, 3, 4, 5, 15, 21, 38, 39, 41 and 49] depict the application of ontologies for knowledge management in Requirement Engineering however the main concern surrounds three main areas (i) handling ambiguity, inconsistency and incompleteness of requirements; (ii) Support for domain knowledge representation to guide requirements elicitation (iii) the use of ontologies to help requirements management. Domain knowledge representation for requirement elicitation is one of focused area of research [3, 5]. Some work highlighted the application of wiki for knowledge management and a study is conducted to extend requirement wiki to semantic wiki [6, 38]. As discussed in previous section, specifically related to requirement validation work [13, 14, 15, 16] indicated the progress in field of validation, some are converting requirement into formal model for purposes of validation, other are using formal model and traceability, yet issues involves are again lot of human effort and minimum accuracy. A few work highlighted the ontological solution here [1, 9] but an again pre-mature model, however no work is addressing requirement validation issues in GSD environment.

In previous section, literature is reviewed in order to analyze various solution of requirement validation, that exists and their limitations, in GSD. This section will discuss the application and use of ontologies for addressing the problem requirement validation in GSD. Many ontologies are suggested for of activities of requirement engineering and few also addresses the problem of requirement validation [1, 2, 3, 5, 15, 21, 38, 39, 41 and 49]. These papers discusses different ontologies that are defining structures of finished artefacts. Moreover they suggests ontological solutions for defining unstructured contents also and to establish

their traces with finished artefacts as well addressing the problem of requirement validation. But no ontological is yet defined for this purpose. Problem of pre-RST is highlighted to address various problems i.e. requirement validation, reusability of requirements etc.

Castañeda, Verónica, et al [3] noted various uses of ontology in area of requirement engineering and highlighted that due to ambiguous requirement, insufficient specification, incomplete specification of requirement, dynamic nature and changing behaviour of requirements, process of elicitation remain inadequate and software cannot meet the expectation of users. These problems of requirement engineering can be addressed by using ontologies. They highlighted three potential area where ontologies can apply and these are (1) structure of requirement definition (2) developing domain ontology (3) developing application ontology, see figure 3. This figure contain three ontologies, however it is extended and not a part of original figure. Further these ontologies can be used for requirement validation. As discussed in previous section pre-RST can address the problem of requirement validation by creating traceability links between finished artefacts and their sources. However in order to automate requirement validation, there should be semantic traceability between target and source artefacts. , Dobson and Sawyer [41] discusses the use of ontologies for address the problem of traceability. Further authors proposed an ontology in which structured requirements are associated on basis of dependency. Riechert et al. [21] defines the semantic structure in order to define relevancy between requirement artifacts. Castañeda, Verónica, et al [3] noted that ontologies represents formal structure and finished contents of requirements, however it requires to develop ontology that also represents unstructured contents and their relationship with finished contents. This paper highlighted roles of stakeholders in capturing requirements and to represent the unstructured contents along with their contributor. Castañeda, Verónica, et al [3] defines the domain ontology and pointed its usefulness for automated requirement validation. However this paper doesn't highlighted how ontologies address the automated requirement validation problem in GSD. Derneval, Diego, et al. [2] in their systematic literature review on application of ontologies, discussed the main phases of requirement engineering process that have been supported by the use of ontologies. According to this review, requirement specification is major phase that is mostly addressed by various research papers whereas research contribution is little in area of requirement validation. Other areas that are addressed are elicitation, analysis & negotiation, and requirement management. Requirement specification is most addressable area because many tools are available that converts informal content into formal contents and lot of research contribution are there on ontologies to specify requirement engineering artefacts. Many contribution are in area of requirement analysis and change management. This paper highlights the resistance in using ontologies when interact with user for validation. Further this paper highlighted the use of ontologies to model some requirement engineering artefacts or activity. An important result drawn in this paper is major use textual representation style to represent requirements. It is difficult to apply semantics on textual requirements as compare to models. Models do not represent true user opinions [2, 40].

Further this paper highlights the usefulness of ontologies in area of requirement engineering however mostly paper provided positive arguments in this favour without empirical evidence. Most of papers are evaluated on basis of experimental nature, yet industrial evidence is lacking. Empirical evidence highlights the usefulness of ontologies to address various requirement problems related to ambiguity, inconsistency and incompleteness of requirements, lack of domain knowledge for guiding requirements elicitation, managing functional and NFR communication of requirements. Ontologies are useful for reducing ambiguity and for automatic error checking and conflict analysis of requirements. It is useful to verify inconsistency and incompleteness of requirements with less effort. Domain knowledge ontologies is useful to guide requirements elicitation process. Using ontologies is effective for manag-

ing requirements changes and non-functional requirements. Use of ontology may facilitate the communication between stakeholders. Empirical evidence also concludes that application of ontologies yet not revealed satisfactory results in GDSD. Many other ontologies are addressing various problem of requirement engineering [1, 2, 3, 5, 15, 21, 38, 39, 40, 41 and 49], and I found only one paper in best of my effort on ontologic solution of requirement validation [1] however this paper is not addressing GDSD.

4. Artifacts ontology

As comprehensively discussed in previous section, in order to address requirement validation, the finished artifacts along with each unit requirements should be electronically indexed and traceable with all unstructured contents to which it belongs [10]. Many solution are proposed for pre-RST among which requirement wikis is useful solution for establishing traces between different finished artifacts of requirements [10, 38]. Use of ontologies are also highlighted for pre-RST and requirement validation. It is also discussed that ontological solution will be useful for automatic requirement validation in GDSD that maintain the ontologies of unstructured contents and maintain semantic relationship between them such that structured contents should be semantically traceable to their sources and origin. However this is a problem of pre-RST; many researchers have contributed in this area however several challenges still exist [10, 16, and 19]. Unfortunately one of the problem of Pre-RST is electronically non-indexing of primary and intermediate artifacts [25]. As pointed out earlier that pre-RST is a problematic area and in GDSD, without maintaining traces of unstructured contents with structured, requirements cannot be traced back and validated from its sources [10]. This raise a serious issue of validation in GDSD when source is not accessible along with its contributor as discussed in previous section.

Figure 3 is showing three ontologies that is requirement, application domain & requirement specification document ontology [3]. Requirement ontology is specification of requirements which is describing the desired functional and non-functional behavior of software in hierarchical way representing structured contents of finished functional and non-functional requirements; application domain ontology defines application domain knowledge and vocabularies, and their relationship; requirement specification document ontology defines semi-structured and structured contents of requirements in form of scenarios, goals and actors. In order to describe traceability between structured and unstructured contents, figure 3 shows an extended framework to an original figure available at [3].

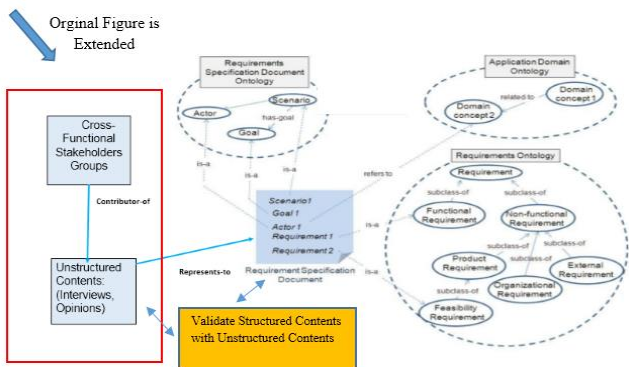


Fig. 3: is extended with Artifact Based ontology

This framework is shows a relationship between scenario, goal, actors and requirements representing structured contents and interviews, emails, blogs, feedback, change notes etc. representing unstructured contents, and cross-functional stakeholders i.e. client, users, marketing group, technical liaisons etc. representing contributors of unstructured contents. Figure 4 shows the abstract model of artifacts ontology.

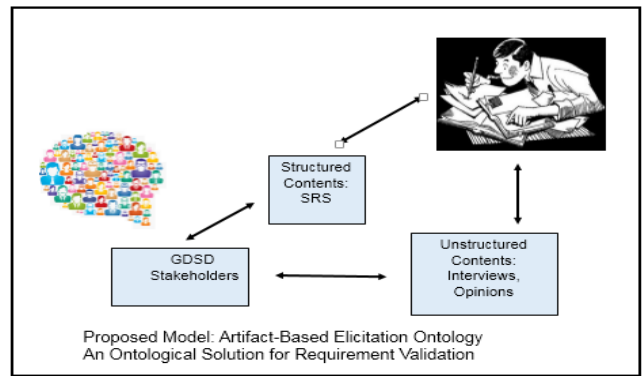


Fig. 4: Abstract model of Artifact Based Ontology

This ontology of unstructured contents describing different types of artifacts that are primary sources and input to elicitation process as well as intermediate sources that are generated during requirement elicitation process, thus all are representing voice of customers. These artifacts are maintained in different format such textual, graphical, audio, and video. Some artifact are original citations while other are transformed, however all are linked with each other as well as with their contributors. This ontology is useful to manage frequent and evolving nature of unstructured contents, along with their contributors in GDSD without losing association. Further finished artifacts is also associated with all unstructured contents. Requirement validation is user oriented process, since all unstructured contents are also representing voice of users.

This ontology can semantically locate the source of finished artifacts thus an ease for to locate the source in order to validate in absence of their contributors. Moreover contributor can also be semantically located when required source is not expressing voice of customer or user. Use of elicitation artifact-based ontology that will manage this evolving and diverse knowledge sources will help in validation of requirement semantically in GDSD in absence of stakeholders, thus making collocating environment. Such an ontology is useful for managing primary, intermediate and finished artifacts however if managed properly, becomes useful for requirement validation in GDSD. Figure 5 shows ontological model describing conceptualization of evolving artifacts and their relationship.

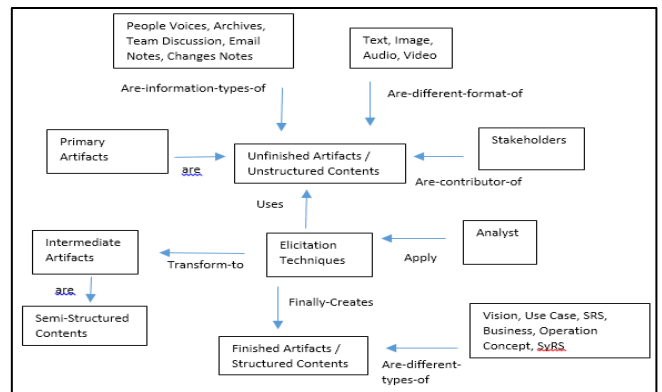


Fig. 5: Ontological Model: Artifacts Evolution and their relationship during Elicitation

5. Application area

This research contribution will reduce the communication and collaborative overheads. Requirements will be validated semantically in absence of stakeholders in GDSD. A step towards collocating the distributed stakeholders, which will help in addressing many GDSD issues. This paper is good contribution for addressing pre-RST issue. It guide requirement engineers to learn the domain vocabulary moreover make his work automated. This further assist in coping incompleteness, ambiguity and incon-

sistency in specification. It will increase the re-usability of requirements.

6. Conclusion

This paper introduced the concept of requirement validation in GSD. This paper raised problem of requirement validation & verification in GSD environment. Moreover a thorough review is taken on automated tools developed for requirement validation. This paper discussed different ontological solutions of requirements as well as addressed requirement validation gaps also. This paper identified use of ontologies in Requirement Validation for GSD. Finally we presented an abstract idea for the proposed solution in form of artifacts ontology. In the last but not least, an important problem of pre-RST is also addressed in this paper.

7. Future direction

Proposed ontological model is a useful contribution to validate requirement in GSD environment, in absence of stakeholders. However this model doesn't define any semantic mechanism that will list valid and invalid requirements automatically, thus making fully semi-automated process. This is a future direction to fully automate the requirement validation process in which some mechanism will evaluate the structured artifacts with unstructured contents and automatically highlights those requirement expression that are invalid. Future work should also focus on development of automated tool for maintaining pre-RST in order to manage evolving nature of artifacts.

References

- [1] Kof, Leonid, et al. "Ontology and Model Alignment as a Means for Requirements Validation." ICSC. 2010.
- [2] Dermeval, Diego, et al. "Applications of ontologies in requirements Engineering: a systematic review of the literature." Requirements Engineering (2015): 1-33.
- [3] Castañeda, Verónica, et al. "The use of ontologies in requirements engineering." Global journal of researches in engineering 10.6 (2010).
- [4] Castañeda, Verónica, Luciana C. Ballejos, and Maria Laura Calusco. "Improving the Quality of Software Requirements Specifications with Semantic Web Technologies." WER. 2012.
- [5] Farfeleder, Stefan, et al. "Ontology-driven guidance for requirements elicitation." The Semantic Web: Research and Applications. Springer Berlin Heidelberg, 2011. 222-226.
- [6] Decker, Björn, et al. "Self-organized reuse of software engineering knowledge supported by semantic wikis." Proceedings of the Workshop on Semantic Web Enabled Software Engineering (SWESE). 2005
- [7] Sommerville, Ian, and Gerald Kotonya. Requirements engineering: processes and techniques. John Wiley & Sons, Inc., 1998.
- [8] Westfall, Linda. "Software requirements engineering: what, why, who, when, and how." Software Quality Professional 7.4 (2005): 17.
- [9] Al Balushi, Taiseera Hazeem, Pedro R. Falcone Sampaio, and Pericles Loucopoulos. "Eliciting and prioritizing quality requirements supported by ontologies: a case study using the ElicitO framework and tool." Expert Systems 30.2 (2013): 129-151.
- [10] Gotel, Orlena, et al. "Traceability fundamentals." Software and Systems Traceability. Springer London, 2012. 3-22.
- [11] Botzenhardt, Achim, Andreas Witt, and Alexander Maedche. "A Text Mining Application for Exploring the Voice of the Customer." AMCIS. 2011.
- [12] Sommerville, Ian, Peter Sawyer, and Stephen Viller. "Viewpoints for requirements elicitation: a practical approach." Requirements Engineering, 1998. Proceedings. 1998 Third International Conference on. IEEE, 1998.
- [13] Sutcliffe, Alistair G., et al. "Supporting scenario-based requirements engineering." Software Engineering, IEEE Transactions on 24.12 (1998): 1072-1088.
- [14] Maiden, Neil A. M. "CREWS-SAVRE: Scenarios for acquiring and validating requirements." Automated Software Engineering 5.4 (1998): 419-446.
- [15] Prakash, N., C. Rolland, and B. Pernici. "Use of domain knowledge for requirements validation." Information System Development Process: Proceedings of the IFIP WG8. 1 Working Conference on Information System Development Process, Como, Italy, 1-3 September, 1993. Elsevier, 2014.
- [16] Hallerstede, Stefan, Michael Jastram, and Lukas Ladenberger. "A method and tool for tracing requirements into specifications." Science of Computer Programming 82 (2014): 2-22.
- [17] Van Lamsweerde, Axel. "Goal-oriented requirements engineering: A guided tour." Requirements Engineering, 2001. Proceedings. Fifth IEEE International Symposium on. IEEE, 2001.
- [18] Fairley, Richard E. Managing and leading software projects. John Wiley & Sons, 2011
- [19] Nair, Sunil, Jose Luis de la Vara, and Sagar Sen. "A review of traceability research at the requirements engineering conference re@ 22." Requirements Engineering Conference (RE), 2013 22st IEEE International. IEEE, 2013.
- [20] Yue, Tao, Lionel C. Briand, and Yvan Labiche. "A systematic review of transformation approaches between user requirements and analysis models." Requirements Engineering 16.2 (2011): 75-99.
- [21] Yang, Yanwu, et al. "Towards semantic requirement engineering." Semantic Computing and Systems, 2008. WSCS'08. IEEE International Workshop on. IEEE, 2008.
- [22] Meth, Hendrik, Manuel Brhel, and Alexander Maedche. "The state of the art in automated requirements elicitation." Information and Software Technology 55.10 (2013): 1695-1709.
- [23] Damian, Daniela. "Stakeholders in global requirements engineering: Lessons learned from practice." Software, IEEE 24.2 (2007): 22-27.
- [24] Ambriola, Vincenzo, and Vincenzo Gervasi. "Processing natural language requirements." Automated Software Engineering, 1997. Proceedings., 12th IEEE International Conference. IEEE, 1997.
- [25] Christel, Michael G., David P. Wood, and Scott M. Stevens. AMORE; The Advanced Multimedia Organizer for Requirements Elicitation. No. CMU/SEI-93-TR-12. Carnegie-mellon univ pittsburgh pa software Engineering inst, 1993.
- [26] Konaté, Jacqueline, Abd El Kader Sahraoui, and Gwendolyn L. Kofschoten. "Collaborative Requirements Elicitation: A Process-Centred Approach." Group Decision and Negotiation 23.4 (2014): 847-877.
- [27] Coulin, Chad Raymond. A situational approach and intelligent tool for collaborative requirements elicitation. Diss. Université Toulouse III-Paul Sabatier, 2007.
- [28] Davis, Alan M. Software requirements: objects, functions, and states. Prentice-Hall, Inc., 1993.
- [29] Cheng, Betty HC, and Joanne M. Atlee. "Research directions in requirements engineering." 2007 Future of Software Engineering. IEEE Computer Society, 2007.
- [30] Faily, Shamal, et al. "Requirements sensemaking using concept maps." Human-Centered Software Engineering. Springer Berlin Heidelberg, 2012. 217-232.
- [31] IHMC: CmapTools web site. <http://cmap.ihmc.us> (June 2012)
- [32] Gall, Mark, and Brian Berenbach. "Towards a Framework for Real Time Requirements Elicitation." Multimedia Requirements Engineering, 2006. MERE'06. First International Workshop on. IEEE, 2006.
- [33] Antón, Annie, and Colin Potts. "The use of goals to surface requirements for evolving systems." Software Engineering, 1998. Proceedings of the 1998 International Conference on. IEEE, 1998.
- [34] Sutcliffe, Alistair. "Requirements rationales: integrating approaches to requirement analysis." Proceedings of the 1st conference on Designing interactive systems: processes, practices, methods, & techniques. ACM, 1995.
- [35] Gervasi, Vincenzo, and Bashar Nuseibeh. "Lightweight validation of natural language requirements." Software: Practice and Experience 32.2 (2002): 113-133.
- [36] Coulin, Chad Raymond. A situational approach and intelligent tool for collaborative requirements elicitation. Diss. Université Toulouse III-Paul Sabatier, 2007.
- [37] Stone, Andrew, and Peter Sawyer. "Using pre-requirements tracing to investigate requirements based on tactic knowledge." ICISOFT (1). 2006.
- [38] Syed, Zareen Saba, Tim Finin, and Anupam Joshi. "Wikilogy: Using wikipedia as an ontology." Proc. the Second International Conf. on Weblogs and Social Media. 2008.
- [39] Omoronyia, Inah, et al. "A domain ontology building process for guiding requirements elicitation." Requirements Engineering: Foundation for Software Quality. Springer Berlin Heidelberg, 2010. 188

- [40] Dermeval, Diego, et al. "A systematic review on the use of ontologies in requirements engineering." *Software Engineering (SBES), 2014 Brazilian Symposium on*. IEEE, 2014.
- [41] Dobson, Glen, and Peter Sawyer. "Revisiting ontology-based requirements engineering in the age of the semantic web." *Proceedings of the International Seminar on Dependable Requirements Engineering of Computerised Systems at NPPs*. 2006.
- [42] Sutcliffe, Alistair, and Pete Sawyer. "Requirements elicitation: Towards the unknown unknowns." *Requirements Engineering Conference (RE), 2013 21st IEEE International*. IEEE, 2013.